UvA-DARE (Digital Academic Repository)

Emergent oscillations in evolutionary simulations: oscillating networks increase switching efficacy

Heerebout, B.T.; Phaf, R.H.

DOI

[10.1162/jocn.2009.21205](https://doi.org/10.1162/jocn.2009.21205)

Publication date

2010

Published in

Journal of Cognitive Neuroscience

[Link to publication](#)

Citation for published version (APA):

Heerebout, B. T., & Phaf, R. H. (2010). Emergent oscillations in evolutionary simulations: oscillating networks increase switching efficacy. *Journal of Cognitive Neuroscience*, 22(5), 807-823. <https://doi.org/10.1162/jocn.2009.21205>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

Emergent Oscillations in Evolutionary Simulations: Oscillating Networks Increase Switching Efficacy

Bram T. Heerebout and R. Hans Phaf

Abstract

■ LeDoux [LeDoux, J. E. *The emotional brain*. New York: Simon & Schuster, 1996] motivated the direct route in his dual-pathway model by arguing that the ability to switch rapidly between different modes of behavior is highly adaptive. This motivation was supported by evolutionary simulations [den Dulk, P., Heerebout, B. T., & Phaf, R. H. A computational study into the evolution of dual-route dynamics for affective processing. *Journal of Cognitive Neuroscience*, 15, 194–208, 2003], in which foraging agents, controlled by simple inheritable neural networks, navigated an artificial world while avoiding predation. After many generations, a dual-processing architecture evolved that enabled a rapid switch to avoidance behavior when a predator appeared. We added recurrent connections to a new “context” layer in the indirect pathway to provide the agents with a work-

ing memory of previous input (i.e., a “context”). Unexpectedly, agents with oscillating networks emerged that had a much higher fitness than agents without oscillations. Oscillations seemed to have similar effects on switching speed as the dual-processing architecture, but they enhanced switching efficacy to a much larger degree. There has been much neurobiological speculation on the function, if any, of neural oscillations. These findings suggest that the facilitation of switching behavior is a likely candidate. Moreover, the strongly improved adaptation in the simulations contradicts the position that neural oscillations are merely a by-product of cell firing and have no functional value [Pareti, G., & De Palma, A. Does the brain oscillate? The dispute on neuronal synchronization. *Neurological Sciences*, 25, 41–47, 2004]. ■

INTRODUCTION

Evolutionary simulations provide an opportunity to investigate the selective pressures that shaped the functional brain architectures. Because the brain does not fossilize, the evolutionary history of the brain is mostly investigated indirectly using comparative methods (cf., Northcutt & Kaas, 1995). Computational simulations of the evolutionary process can, however, provide more direct insights into which functional mechanisms and neural architectures arise under specific environmental conditions. We argue that if a particular model emerges easily (i.e., with few environmental constraints or under different conditions), an analogous model is likely to have also evolved during biological evolution. In addition, evolutionary simulations may not only strengthen existing models but may also produce novel architectures that perform tasks in ways that have not yet been explored. Here, we report investigations of the adaptive value of recurrent connections in the artificial neural networks of evolving simulated agents. We extended earlier work (den Dulk, Heerebout, & Phaf, 2003) on the evolutionary justification of LeDoux’s (1996) dual-pathway model by including recurrent connections to a new layer in the indirect pathway (see Figure 1). The simulated networks controlled agents, which had to navi-

gate a virtual environment to find food and avoid predation. Surprisingly, the recurrent connections not only led to a major increase in the agents’ success but also simultaneously induced oscillations of activations in the network, which were not previously observed.

Computer simulations allow researchers to perform experiments that would otherwise be impossible due to time, financial, or ethical constraints (Peck, 2004). To mimic the evolutionary process, a genetic algorithm (Holland, 1975) was applied to agents performing tasks in a virtual environment (cf. Beer, 1990). Goldberg (1989; p. 1) defined genetic algorithms as “search algorithms based on the mechanics of natural selection and natural genetics”. Similar to DNA in biological systems, coded parameter sets map to individual solutions to a problem in genetic algorithms. Starting from an initial population (e.g., consisting of individuals with random parameter values), optimal solutions are searched through reproduction and variation. The fitness is defined as a measure of how well the parameter set solves the problem. Fitter solutions have a higher chance of survival and reproduction. During reproduction, the parameters are liable to mutations and crossovers. Mutation occurs randomly, with low probability, and, for instance, flips single bits from 0 to 1 or from 1 to 0. Mutation can thus improve performance by occasionally suggesting a new, and fitter, partial solution. With crossover, two individual solutions selected for reproduction are combined. Because

Figure 1. Schematic view of the network's initial architecture. After the control simulation, a "context layer" (in gray), allowing for recurrent processing from the hidden layer, was added.

a higher fitness gives a higher change of survival and reproduction, the overall fitness of the offspring is generally higher than the fitness of the parents. After many repetitions (i.e., generations) the population can gradually evolve toward optimal solutions.

Evolutionary simulations may be useful to cognitive neuroscience in three different respects. First, if a set of environmental conditions leads to a particular feature, it confirms the possibility of it having evolved gradually with minimal changes between generations. Nilsson and Pelger's (1994) simulation of the evolution of the eye, for instance, countered the famous remark by Bishop Paley that surely the human eye could not have evolved gradually. Second, if a general neural architecture is likely to emerge in these simulations (e.g., despite variations in the environmental conditions), it is also likely to have evolved in nature. Therefore, when comparing alternative accounts, the neural architecture emerging from evolutionary simulations should be favored over other models that explain the data equally well, but are lacking evolutionary support. Finally, evolutionary simulations may produce novel architectures not considered before by cognitive neuroscientists. Simulations of the evolutionary process may occasionally serve to synthesize new models, without the intervention of the human researcher. Thus, it can be seen as a kind of synthetic psychology (Braitenberg, 1984; cf., Dawson, 2002; Evans & De Back, 2002), which starts with the evolution of neural architectures and only afterward investigates the full range of behavioral consequences.

An innovative aspect of LeDoux's (1986, 1996) approach was that his evolutionary reasoning focused on internal processes and neural architectures, instead of on externally observable behavior, as had been often the case in ethological approaches (Lorenz, 1966; Darwin, 1872/1965). LeDoux suggested that activations due to affective stimuli can travel from the sensory thalamus to the amygdala via two parallel pathways: one directly from the sensory thalamus, the other indirectly by running through cortex before reaching the amygdala. The direct pathway is faster, has less capacity, and is more coarsely grained than the indirect pathway. LeDoux gave an evolutionary justification of the dual-pathway architecture by arguing

that in threatening situations the evolutionary cost of a miss exceeds the sum of the costs of the many false alarms produced by the short pathway. When there is more time to process stimuli extensively, the costs of the false alarms may even be reduced through inhibition of the direct action tendencies via the long pathway. LeDoux's formulation in terms of evolutionary fitness clearly facilitated the translation into simulations with genetic algorithms.

Den Dulk et al. (2003) largely followed the simulation setup of Beer (1990), who investigated the evolution of artificial neural networks that controlled the foraging behavior of agents. The agents could detect plants using chemotaxis (i.e., they detected the scents emitted by the food). The weights of the networks' connections were encoded into the agents' genes. Den Dulk et al. introduced a temporal factor in the simulation by implementing time delays due to the transmission of activation by a node. Processing in the biological nervous systems always consumes time, and time plays a crucial role in LeDoux's evolutionary reasoning. Predators were also added to Beer's setup and the agent had to discriminate their scents from the food scents (see Figure 2). The agents' reproductive success and survival (i.e., their fitness) depended both on their ability to avoid predators and their success in gathering plants.

After 1000 generations of simulated evolution, agents with dual-processing networks resembling LeDoux's architecture developed. Processing in the direct pathway induced fast avoidance of both predator and plant, whereas slower indirect processing led to plant approach and facilitated predator avoidance. These agents were only found in simulations when the scents of plant and predator were hard to distinguish and the fitness (i.e., "lifetime" multiplied by the total number of plants collected) reflected time pressures in escaping from the predator. The qualitatively different types of processing in the two pathways were further supported by lesion studies (of the separate pathways), which are performed more easily in models than in animals. Undoubtedly, the actual mammalian fear network is much more complicated than the network model that emerged from these simulations. Nevertheless, the converging neurobiological and computational results provide strong support for LeDoux's general architecture with parallel processing pathways differing in time costs and complexity.

The networks of den Dulk et al. (2003) only had feed-forward connections (e.g., Haykin, 1999), but can easily be extended with feedback, recurrent, connections. Fully recurrent and synchronous network models have been pioneered by Hopfield and Tank (1986), but we aimed at a different type of recurrent network. In our setup, the activation transfer delays meant that specific layers of nodes could maintain activations from previous time periods. So we aimed at obtaining a working memory capacity (e.g., Baddeley, 1986) by allowing recurrent connections to evolve to and from an additional layer of hidden "context" nodes (for a similar architecture, see

Figure 2. A schematic depiction of the artificial world with agent, predators, and plants. The lines from the center to the front indicate the agent's and the predators' directions. The magnification shows an agent's network from the control simulation.

Elman, 1990). The context nodes could retain previous activations of the original hidden layer and, subsequently, provide a context for processing in this hidden layer (e.g., see also Ashby, Ell, Valentin, & Casale, 2005; Phaf, Mul, & Wolters, 1994). The interactions between hidden and context layers should enable the temporary maintenance in working memory of previous stimulus objects in the environment, and possibly even the planning of future events (see Schacter, Addis, & Buckner, 2008). For instance, the agent may now be able to predict where a predator is going, or keep a plant's location in "mind" while avoiding a predator.

To investigate the potential of a working memory function in a recurrent architecture, we performed several exploratory simulations with a range of parameter settings, which yielded variable results due to the very large search space. One simulation, however, showed a sudden jump in fitness to a level more than two times higher than found in the simulations of den Dulk et al. (2003). Additional analyses of this simulation revealed an alternative dual-processing architecture, with oscillatory network activations and enhanced functional capabilities. We will first describe the general simulation method in more detail, and present a replication of the den Dulk et al. simulation for comparison purposes with the new simulations.

METHODS

Simulation Setup

The general simulation setup of den Dulk et al. (2003) was followed. The networks with the new recurrent connections were introduced only after running the control simulation. The virtual world (see Figure 2) for the agent, plants, and predators had a size of 400×400 arbitrary length units, and was torus-shaped (i.e., had no boundaries). The agent, plants, and predators all had the same circular shape and size (i.e., a radius of 10 length units). The agents and predators moved by exerting force with their motor actuators, located on both sides of their body.

When one actuator exerted a greater force than the other, the agent deflected to the side of the lesser force. Discrete time steps (i.e., iterations) and floating-point variables were used to approximate continuous two-dimensional movement over time. The laws of classical mechanics (e.g., see Francis, 1973) were implemented to specify speed and acceleration (for both translation and rotation) as functions of the motor actuators' force and the agents' inertia and friction while moving (linearly dependent on speed). Movement costs the agent a small amount of energy proportional to the force exerted (a maximally activated output node resulted in a consumption of 0.001 energy unit per time step).

All entities emitted virtual scents equally in all directions of the two-dimensional world, hence, intensity decreased quadratically with distance to the source. Agents and predators detected the scents with olfactory sensors, located to their left and right at an angle of 45° from the front. Activation of the sensors depends on the distance to the source so they are differentially activated if the source is to the right or left of the agent. The agents emitted only one type of scent and the predators were preprogrammed to move into the direction of its source. The predators and plants, however, emitted two different types of scents, Scent A and Scent B (i.e., predators emitted 1A and 0.5B, plants 0.5A and 1B; see Appendix A).

An agent's network (see Figure 1) processed the sensory input and produced output to the motor actuators. The input layer consisted of four nodes. Two nodes received input from the left A and B sensors and the other two from the right A and B sensors. The input layer connected directly to the output layer and to the hidden layer. The four-node hidden layer gave off connections to the output nodes. All weights (between -10 and 10) were encoded into the agent's genes and started out at zero in the first generation. The weights could change between generations, but were constant within an agent's "lifetime" (i.e., the agents could not learn). The connections were symmetrical so that the weights on the left side mirrored the right side. This halved the search space for

the genetic algorithm and also made sense behaviorally. Because of the mirrored connections, if activation of a particular hidden node caused the agent to turn left, then its mirrored counterpart would turn the agent right when activated. The first two hidden nodes are considered to be located on the agent left side and the second two on the agent's right. Hence, H_1 mirrors H_4 and H_2 mirrors H_3 . For the calculation of a node's activation standard rules applied (see Appendix B). The sum of the weighted activation of the sending nodes, plus a fixed bias of 0.1, was squashed resulting in a new activation between 0 and 1 (Rosenblatt, 1962). We took the time required for the transfer of activation by a node as the basic time unit and called this a time step or iteration.

The genetic algorithm procedure (Michalewicz, 1992) selected agents to create modified offspring. The individuals were first "tested" in 12 virtual worlds to assess their skills of avoiding predators and collecting plants. The worlds always contained 6 predators and 10 plants with different random initial positions. Each agent remained in the world until either a predator made contact with the agent, or a maximum of 10,000 time steps had passed. When the agent made contact with a plant, the plant was considered eaten. The plant disappeared from the world, but a new plant would appear instantaneously at a new position, keeping their number constant. The energy from the eaten plant (1 energy unit) was transferred to the agent. The number of time steps an agent spent in a particular environment was multiplied by the amount of net energy the agent had at the moment the agent was removed from the world. The average over the 12 tests defined the agent's fitness, which determined its chance of reproduction and the chance of remaining in the population (see Appendix C). The next generation

started out with the offspring and the remaining agents. The energy level of the offspring is set to five energy units upon creation. The initial population consisted of 18 individuals. Later on, the population size would vary, depending on the relative fitness of the individuals.

Offspring was created by copying the genes of two selected agents. There was a 50% chance that a crossover occurred between the copies. Each copied gene, encoding a different connection weight in the agent's network, was mutated slightly. The mutations were randomly drawn from a Poisson distribution (the average weight change, in absolute value, was 0.13, with connection weights ranging from -10.0 to 10.0), ensuring that small mutations were more likely to occur than large mutations. Simulations ran for 10,000 generations.

RESULTS

Control Simulation

The first simulation served as a control simulation and a replication of the results of den Dulk et al. (2003). The fitness development over the generations (Figure 3) shows an increase in the first 500 generations to about 3000, and after a peak between generation 550 and 600, it returned to this value. The fluctuations were caused by the noise in the fitness test and the mutations and crossovers in the genes. The test can only estimate the skills of foraging and predator evasion with a limited accuracy. Moreover, the mutations were intended to cause variation for the search algorithm, and thus, lead to fluctuations in measured fitness. The fitness did not increase after the first 600 generations and the agents' networks did not seem to develop any further in the control simulation.

Figure 3. The fitness (in Energy units $\times$ Time steps spent in the environment) development as a function of generation (averaged over clusters of 50) in the control simulation and in the subsequent simulation with recurrent connections to a context layer.

Figure 4. Schematic view of the average network in the last generation resulting from the control simulation. The nodes from the three layers are coded with I (input), H (hidden), and O (output), respectively. The subscript of the I-nodes denotes the type of receptor (Scent A or B) and their position L (left) or R (right). The H-nodes are numbered with each even-numbered node mirroring its odd counterpart. The subscript of the O-nodes denotes the motor actuator (left or right) it projects to. The thickness of the connections indicates the weight category (either in the range [2.5, 5.0], [5.0, 7.5], or [7.5, 10.0]). Connections with weights between -2.5 and 2.5 are omitted because their effect is deemed negligible. Excitatory connections are represented by solid arrows and the inhibitory connections by dotted ovals.

To analyze the evolved connection structure, the weights in the networks were averaged over all agents in the last generation. Because mutations were small, the individuals within one generation were not very different. This allowed the “average agent” of one generation to serve as a repre-

sentative agent for that generation. The average agent’s network (see Figure 4) nicely illustrates the dual-processing architecture. The connections of the direct pathway always cause avoidance of both stimuli. When a plant (mostly emitting scent B) is detected, the parallel excitatory connections from the “B” input nodes cause greater motor activation on the side of the plant. The agent thus moves away from the plant. This behavior is amplified by the crossed inhibitory connections from the “B” input nodes. The inhibitory connections from the “A” input nodes also run in parallel. This dampens, but not fully reverses, the avoidance when the agent encounters a predator (mostly emitting Scent “A”). Processing via the hidden nodes is more elaborate and, in contrast to processing in the direct route, differentiates between the two types of stimuli. The hidden nodes H_1 and H_2 activate the left output node (causing a right turn). Their mirrored counterparts, H_3 and H_4 , cause the agent to turn left. Thus, when, for instance, a predator approaches from the left, the “A” input node activates both H_1 and H_2 moving the agent to the right (avoiding the predator). When a plant is detected on the left, however, H_4 is activated with H_2 and so the right motor actuator reaches a higher activation causing an approach response to the plant’s location. Through the indirect processing, the network is thus able to produce a differential response and reverse, or enhance, the general avoidance tendency resulting from the direct activation.

Computer simulations allow for an analysis of both internal processing and of external behavior. The agent controlled by the above network was confronted with plants and predator stimuli at a fixed position, 45° left of the agent. The agent’s orientation was subsequently measured for five consecutive time steps (see Figure 5). To investigate the processing in the two pathways separately, the measurements were repeated after lesioning either the

Figure 5. The orientation of the averaged agent from the last generation of the control simulation over five time steps when the agent was confronted with a plant or a predator placed at a positive angle of 45° relative to the agent. The agent was tested with its network intact (left), direct connections only (middle), and indirect connection only (right). If the agent’s orientation is positive, the agent moves toward the stimulus, and if the agent’s orientation is negative, the agent turns away from the stimulus.

direct or the indirect pathway. In the “direct” condition, all connections from input to output via the hidden nodes were set to zero and in the “indirect” condition the same was done with all direct connections from input to output.

The agents showed no response in the first time step because the transfer of activation to the output nodes takes at least one time step. At T2, the first activation to reach the output nodes caused the intact agent to turn away from both plants and predators. The agent turned away further from the predator at T3, when the activation from the hidden nodes also reached the output nodes. With a plant stimulus, however, the agent started to reverse its initial response at T3. After T3, the agent strengthened its avoidance behavior from the predator, or its approach behavior toward the plant. The results with the lesioned networks support these conclusions. The agent with only “direct” connections swiftly moved away from all stimuli. The “indirect” agent correctly differentiated between plant and predator, but it was slower to respond, only starting at T3. A simple form of dual processing, as postulated by LeDoux (1986, 1996) for the mammalian brain, thus, also seems to have emerged in this simulation.

A Sudden Jump in Fitness

We aimed at the development of a working memory capacity for previously processed stimuli by adding a context layer of four nodes (see Figure 1). The new nodes had connections, which were also coded in the agent’s genes, to and from the hidden layer. The activations stemming from the hidden nodes were sent back in the following time step, so that previous stimuli could influence current processing. The fitness development for agents with this architecture initially resembled that of the control simulation, but showed a sudden jump in fitness around generation 7500 (see Figure 3). The doubling in fitness (i.e., from about 3000 to 6000) indicated a qualitative change in the agents’ weight configuration that had a major impact on their performance. Contrary to our initial expectations, the jump did not result from the development of some working memory capacity, but coincided with the emergence of oscillations in the networks (see Figure 6). A schematic view of the average network of the agents in the last generation is shown in Figure 7.

To analyze the interactions of the different layers, the nodes’ activations were recorded, when the agent was confronted with a plant (see Figure 6A) and with a predator (see Figure 6B), both at 45° to the left of the agent. With a food stimulus the network showed steady oscillating activations with a period of two time steps. When the plant was on the agent’s left, H₁ was activated more strongly than H₂, which resulted in an excitation–inhibition exchange between H₁ and C₃. Because C₃ not only inhibited H₁, but also activated H₃ and H₄, the latter nodes also oscillated. The excitation of H₄ was, however, stronger

than that of H₃, which caused a turn to the left. Behaviorally, the oscillations in the output layer led to pulsating movements by the agent toward the plant. In contrast, the context nodes were not activated by the predator stimulus so that predator activations and avoidance behavior were not modulated by oscillations. Oscillations were only found when the input of Scent B was stronger than that of scent A. Hence, due to these “food oscillations,” the agent zigzagged toward plants, but, in contrast, moved away from predators in a straight line. Instead of temporarily storing activations of the hidden layer from the previous iteration, the recurrent connections between hidden and context layers act as a generator of oscillations, which propagate to the output layer through the feed-forward connections.

The oscillations are produced by the interactions between hidden and context layers. The hidden nodes H₂ and H₃ activated C₃ and C₂, respectively, but H₂ and H₃ were, in turn, inhibited by C₃ and C₂. This corresponds to a “flip-flop” mechanism, in which excitation and inhibition alternate. Interestingly, from a neurobiological perspective (Ritz & Sejnowski, 1997), a similar setup of recurrent inhibition has been identified as one of the possible generators of gamma oscillations (20–70 Hz), which are presumably involved in object perception. Biological neural networks show a much higher level of temporal differentiation than simulated networks due to the high diversity of inhibitory GABAergic interactions (Klausberger & Somogyi, 2008). In our synthetic approach with evolutionary simulations, this simple circuit emerged as the basic oscillation generator, giving rise to substantial fitness gains under these environmental conditions. Oscillatory activity was associated more strongly with the perception of plants than of predators. The “flip-flop” configuration was not present between H₁ or H₄ and any of the context nodes. H₂ and H₃ only received activation from the “B” input, which was emitted mainly by the plants. Hence, the oscillations periodically switch off and on the motor actuators when foraging, so that, when a threatening stimulus appears, the switch to avoidance can be made more effectively.

To understand the dual-processing performance of the oscillating agents, we also analyzed the processing via both pathways. Similar to the test in the control simulation, the agent’s orientation was registered after the presentation of plant and predator at a 45° angle. The agent with the intact network approached the plant and avoided the predator. However, the processing in the direct pathway differed from the control simulation. When the indirect pathway was lesioned (see Figure 8), the agents always turned toward both types of stimuli. Only indirect processing resulted in the appropriate avoidance response to a predator stimulus. This “inverse dual processing” does not seem to correspond to the dual-processing dynamics suggested by LeDoux (1996). The oscillations appear to take over some of the functionality of dual processing in the evolutionary simulations.

The Adaptive Function of Oscillations

A core assumption in the evolutionary justification of LeDoux (1996) is that the dual-pathway architecture enables a fast switch in behavior as soon as stimuli that are crucial for survival are detected. We tested this assumption by investigating the agent's movements after an abrupt change of stimulus (see Figure 9A and B). We first presented the agent with a plant, placed at an angle of 45° left of the agent. After the initial approach, the plant was suddenly replaced by an (immobilized) predator at the same position. The agent's speed (the distance covered be-

tween time steps) and angular speed (the difference in orientation between time steps) were recorded in three conditions: without switch, a switch at T8, or a switch at T11. The plant was switched to a predator at different times to check for differences in switching behavior due to the phase of the oscillation.

The agent from the control simulation braked, slightly turned away, and accelerated after the plant was replaced by the predator. The oscillating agent behaved similarly, but the differences in translational and angular speed between switch and no-switch was much larger than for the control agent. The oscillating agent's foraging speed

Figure 6. The recorded node activations sorted per layer when the agent was confronted with a plant (A) and a predator (B), both at 45° left of the agent. The red graphs were used for C_4 , H_3 , O_L , and I_{BR} ; black for C_3 , H_1 , O_R , and I_{BL} ; green for C_1 , H_4 , and I_{AR} ; and blue for C_2 , H_4 , and I_{AR} . With the plant stimulus, the nodes in both hidden layers started to oscillate and this was projected to the output nodes. With the predator stimulus, the nodes in the context layer were not activated and the output nodes reached steady activation levels.

Figure 7. Schematic view of the average network of the last generation of agents with evolved connections from and to the context layer.

was lower, but when the agent detected the predator, its speed quickly increased and it turned away sharply. The sum of the differences in translational and angular speed between the no-switch and switch conditions (i.e., the surfaces between the no-switch graph and the deviating graphs shown) was 6.57 and 0.44 for the control agent versus 38.0 and 0.78, respectively, for the oscillating agent. The phase (T7 or T10) of the oscillation in which the switch occurred mattered only in the first few iterations. When summed over the first five time steps after the switch, the speed differences between switch and no-

switch were almost equal (T7: 1.52; T10: 1.35). Due to the absence of oscillations, no phase differences could occur in the control simulation (T7: 0.18; T10: 0.17). For the angular speeds, phase effects averaged over five time steps (oscillating agent, T7: 0.45; T10: 0.24; control agent, T7: 0.19; T10: 0.18) were somewhat larger, but leveled out later.

Three factors may contribute to the substantial jump in fitness. The agent may live longer because it is better able to escape from predators. The agent may also be able to gather more food. Finally, the increase in fitness could be associated with a decrease in energy consumption due to the slowing down of foraging. Table 1 shows the energy consumption averaged over a time period of 250 generations somewhat before and after the jump in fitness (which occurred around generation 7500). Energy consumption per time step is indeed reduced somewhat (-9.3%). The total energy consumption in the fitness test, however, increased slightly (0.3%) because the agents spent more time in the environment. The reduction in energy consumption can, thus, not account for the fitness jump. Another substantial contribution to the fitness jump is made by the increased ability (12.7%) to escape from predators. The largest contribution to the fitness jump is made by the number of plants eaten (an increase of about 0.6 plant), which appears to be due to the prolonged opportunity for gathering food, and an increased efficacy of foraging. A successful escape from a predator apparently helps foraging by subsequently accelerating the agent's movements toward plants. The ability to reorganize behavior more effectively after a threatening stimulus appears probably accounts for the doubling of fitness. The huge evolutionary benefit of this mechanism apparently overrides the costs of the initial general approach tendency caused in the direct pathway of the oscillating agent.

Figure 8. The orientation of the averaged agent of the last generation of agents with evolved connections from and to the context layer.

Figure 9. The “switch test” results for an agent with the averaged network from the control simulation (A), the oscillating network from the second simulation in which the network only oscillated when the plant was detected (B), the oscillating network with direct processing as described by LeDoux (1996) (C), and the oscillating network with inverse direct processing (D). The solid graphs show the agent’s speed and the dashed graph shows the agent’s angular speed. A positive angular speed indicates the agent turns left (toward the stimulus) and a negative one indicates the agent turns right (away from the stimulus). The black graphs depict the speed and angular speed when the plant was not switched with a predator. The green graphs deviates from the black graph showing the behavior when the switch was made at T7, the red graphs at T11. The two time step delay of the deviations is caused by the time cost of the neural processing.

Table 1. Comparative Results

<i>Dependent Variable</i>	<i>Pre-jump</i>	<i>Post-jump</i>	<i>Relative Change Percentage</i>
Fitness	3438.6	7844.1	128.1%
Energy consumption per time step	0.0030	0.0027	−9.3%
Energy consumption per fitness test	0.649	0.671	0.3%
Plants eaten per fitness test	1.3	1.9	44.4%
Lifetime (iterations)	217.0	244.4	12.7%

Pre-jump = generation 7000–7250; post-jump = generation 7750–7800.

Oscillations and Dual Processing

To investigate whether oscillations could replace the functionality of dual-processing dynamics, we repeated the last simulation seven times. If oscillations have a similar function as dual processing (i.e., in facilitating switching behavior), the development of a classical dual-processing architecture may become less likely. We also expected that, although the initial chance of the evolutionary “discovery” of oscillations may not be very high, the trait would prosper once one individual had acquired it. With the fitness advantages we obtained earlier, oscillating organisms should swiftly win the evolutionary competition at the expense of non-oscillating rivals. To promote the emergence of oscillations and to reduce the chance of getting stuck in suboptimal weight configurations (i.e., only a local optimum), the simulations were now allowed to run for 20,000 generations. The simulations were exact replications of the previous simulation, but new pseudorandom numbers were, of course, used in the stochastic decisions (e.g., the weight mutations).

Oscillating networks emerged in all but one of the simulations, and the oscillating agents again reached very high fitness levels. The simulations yielded oscillating agents both with “classical dual processing” ($n = 30$) and with “inverse dual processing” ($n = 67$) dynamics. The weights of the agents from the separate simulations could not be averaged because different hidden–context node combinations were responsible for the oscillations. To illustrate the findings in the simulations, representative high-fitness agents from the last generation were selected and analyzed with the same methods as in previous simulations.

Classical Dual Processing

The “classical dual-processing” agent showed initial avoidance for both types of stimuli (see Figure 10A), and a later differentiation of plant and predator. The emergence of oscillations can be seen from the activation recordings when the agent was confronted with both types of stimuli (see Figure 11A). The “flip-flop” interaction between hidden and context nodes again caused the oscillations. In contrast to the first oscillating network (see Figure 6), the oscillations occurred here with both plant and predator and the output nodes oscillated out of phase. When the agent, for instance, detected the plant, both output nodes oscillated, but the output node on the opposite side of the plant had the highest amplitude, so that the agent turned toward the plant. For the predator, the output node on the side of the predator had the highest amplitude, resulting, of course, in avoidance. In addition, the troughs in the “predator” oscillations were higher than in the “plant” oscillations, which allowed the agent to gather sufficient speed while moving away from the predator. The period of the oscillation cycle, six time steps, was also longer than in the previous

simulation. As a consequence, the output nodes peaked for two time steps and were inhibited for two time steps. The lower frequency of oscillations, however, did not seem to lead to qualitatively different behavior.

The “classical dual-processing” agent exhibited oscillatory output with both stimuli. To investigate how this influenced switching efficacy, we again measured the agent’s performance in the no-switch, the switch at T8, and the switch at T11 conditions (see Figure 9C). This agent showed the largest behavioral changes after the switch (the total sum of the differences in speed and angular speed between the no-switch and switch conditions, i.e., the surface between the graphs, was 38.3 and 1.35). The sinusoid form of the angular speed indicates that the agent initially made large turns while zigzagging toward the plant. The oscillations had a smaller amplitude while avoiding the predator than while foraging.

Inverse Dual Processing

The “inverse dual-processing” agent showed approach to both plants and predators as a consequence of direct processing (see Figure 10B). Similar to classical dual processing, response differentiation occurred only in the indirect pathway. The “inverse dual-processing” architecture did not show up in non-oscillatory networks, where it would presumably correspond to much lower fitness levels than classical dual processing. The fitness levels of the oscillatory agents, both with inverse and classical dual processing, were, however, similar.

The node activations (Figure 10B) revealed that the “inverse dual-processing” network also oscillated with both types of stimuli. The period of the oscillations differed for the two types of stimuli. For the plant, the period was two time steps, whereas it was six time steps for the predator. Plant oscillations also had a much higher amplitude than predator oscillations. When the agent detected the plant, the output activations alternated almost between zero and one. This allowed the agents to turn sharply when it detected a plant. With the predator, the steady high levels of activation and the low amplitude oscillations ensured the agent gained sufficient speed to flee.

Switches from plant to predator in the “inverse dual-processing” network (Figure 9D) also resulted in much more pronounced speed changes (the sums of the differences in translational and angular speed between the no-switch and switch conditions over the interval shown were 33.2 and 1.71) than in the non-oscillatory network. From the angular-speed graph, it can be seen that the agent followed a widely curving trajectory (i.e., making large turns) while approaching the plant. The oscillation was, however, almost compressed to a straight line when the agent detected the predator.

Both the inverse and classical dual-processing agents with oscillating activations showed much more adaptive behavior than the classical dual-processing agent without oscillations. To analyze the specific contribution of the

direct route, we also investigated switching behavior after lesioning the direct route. In the control simulation (Figure 12A), the lesioned agent showed a slower, but more pronounced, response to the switch (the difference sum between the no switch and switch graphs increased relative to the intact network to 32.8 and 1.10 for the translational and angular speed, respectively). In the oscillating agents, the lesion had almost the opposite effect. These agents only reacted minimally to the switch, particularly with respect to translational speed. The lesioned “food oscillation” agent (Figure 12B) and the “classical dual processing” agent (Figure 12C) hardly moved and only managed to turn away from the predator. The “inverse dual-processing” agent initially moved slowly toward the plant and was unable to change its speed after the switch to the predator. The direct route, therefore, seems to have an energizing

role in the oscillating network. The direction of the initial direct response (avoidance in the classical and approach in the inverse architectures) seemed to have a negligible effect on the final performance of the oscillating agents. Direct processing, however, apparently exerts an important preparatory influence on motor behavior. Due to this adaptive function, which differs from the function of the indirect pathway, the direct route developed in all oscillating agents.

In sum, all oscillating agents showed greater efficacy in switching behavior than the non-oscillatory agent. Particularly the food oscillations, increased the susceptibility for interruptions due to threatening stimuli. The last two agents, for instance, made wide turns while foraging, as if they hesitated and were on their guard for predators. The adaptive value of switching abruptly, and vigorously, from approach to avoidance likely explains the emergence of

Figure 10. The orientation over time of a “classic dual-processing” agent (A) and an “inverse dual-processing” agent (B) when confronted with a plant or a predator.

Figure 11. The oscillating node activations of a “classic dual-processing” agent (A) and an “inverse dual-processing” agent (B) when confronted with a plant or a predator.

oscillating networks in all, but one, of the last series of evolutionary simulations.

DISCUSSION

The control simulation confirmed that a feed-forward dual-processing architecture, as has been proposed by LeDoux (1996), is an adaptive model when there is time pressure to differentiate approach and avoidance responses. The addition of recurrent connections to this architecture unexpectedly produced adaptive oscillatory networks. At the precise moment of the emergence of oscillations, fitness values more than doubled relative to the non-oscillatory control simulation. Further simulations consistently produced agents with oscillating networks.

The oscillations in our simulations appear to take over the functionality of the type of dual processing as suggested by LeDoux, and even seem to make the direct route superfluous. Lesioning the direct route in these networks, however, reduced the ability of the agent to flee

from predators (i.e., particularly with respect to its translational speed). The indirect route discriminates the different types of stimuli and sets up appropriate action tendencies, but the direct route seems to be crucial to energize these tendencies and to convert them into actions.

Oscillating networks with classic and inverse dual processing had similarly high fitness levels. This suggests that the direct route is not solely involved in fast avoidance responses, but has a more general response enhancing effect. Phelps and LeDoux (2005) already argued that the study of the amygdala (and its role in dual processing) may have been biased toward aversive emotional processing because they are more often investigated, mostly in animal fear-conditioning studies, than positive, appetitive, emotions. Recently, a number of human neuroimaging studies into the processing of positive and negative stimuli have also supported the involvement of the amygdala in both valences. Cunningham, Raye, and Johnson (2004), for instance, found that amygdala activation primarily correlated with emotional intensity, irrespective of

valence, when making implicit and explicit evaluations of emotional stimuli. In line with these findings, our simulations suggest that dual processing is preserved in oscillating networks but that direct processing is no longer associated solely with avoidance.

Biological neural networks exhibit oscillations in a number of different frequency bands, which may serve different functions (see Buzsáki & Draguhn, 2004). The simplifications required by a simulation model make it impossible to identify the frequency band corresponding to the emerging oscillations with any certainty, but the function of these oscillations may provide important clues. Ritz and Sejnowski (1997), for instance, suggested

that gamma oscillations (20–70 Hz) are involved in object perception (cf. foraging by the agent). The sensorimotor role of gamma oscillations is, moreover, supported by a study of Rougeul-Buser and Buser (1997). They observed 40 Hz oscillations in the motor, parietal, and visual cortices when a cat was waiting in front of a hole in the wall from which at times a mouse could pop out and then quickly disappear. When the cat was simply watching the mouse in a Perspex box, however, lower frequencies of 10–15 Hz showed up. We would argue that the cat in the former situation has prepared itself to quickly switch from immobility to vigorous attack, whereas in the latter situation, where the cat cannot reach the mouse, both

Figure 12. The “switch test” results repeated with all agents with their direct connections lesioned. The graphs of the control simulation stop premature in both switch conditions because the lesioned agent is not able to evade the predator. After the agent collides with the predator, it is removed from the environment. All oscillating agents show practically similar speeds in all conditions and, therefore, only the black continuous line of the no-switch condition is visible.

the need for this preparation and the corresponding oscillation frequency may be lower.

Low-frequency oscillations in the theta band (4–10 Hz; Buzsáki & Draguhn, 2004) have been observed in the amygdala during the anticipation of noxious stimuli (with cats; Paré & Collins, 2000), and when confronted with conditioned fear stimuli (with mice; Seidenbecher, Laxmi, Stork, & Pape, 2003). Because in our simulations the oscillations were more often associated with appetitive, food stimuli than with aversive, predator stimuli, it seems unlikely that the emerging oscillations correspond to theta rhythms. In our simulations, the oscillations were limited to the indirect route, which primarily has a sensorimotor role by differentiating between stimuli and setting up different action tendencies. Because switching speed increases with frequency, the evolved oscillations should have a high (i.e., gamma) frequency. Low-frequency oscillations are able to recruit very large networks in neuronal space (Buzsáki & Draguhn, 2004), and can, therefore, easily exert modulatory functions in other areas. Both Seidenbecher et al. (2003) and Paré and Collins (2000), indeed, noticed amygdalo-hippocampal synchronization of theta rhythms, which they argue facilitates both the laying down and retrieval of fear memories.

The precise function of oscillations in the mammalian brain has been subject to an extensive debate in the scientific literature and a wide range of different functions has been put forward, such as the binding of cell assemblies (Gray, König, Engel, & Singer, 1989), input selection (Hutcheon & Yarom, 2000), consolidation and combination of memories (Reyes, 2003), representation by phase information (Buzsáki & Draguhn, 2004), selective amplification (Lengyel, Huhn, & Érdi, 2005), and sequence learning (Ulanovsky & Moss, 2007). Our findings agree well with a suggestion of Schaefer, Angelo, Spors, and Margrie (2006), who argued for an enhanced capability of oscillating neural systems to discriminate between stimuli. They are also in line with a proposal by Fries, Nikolic, and Singer (2007), who described the facilitation and acceleration of a winner-take-all mechanism by gamma oscillations. In their view, the interaction between excitatory pyramidal neurons and inhibitory interneurons results in a time-critical competition. Only the few pyramidal cells that are able to spike sufficiently early in the gamma cycle are able to spike at all. In this manner, the weaker cells are suppressed, which could improve the signal-to-noise ratio (see also Brody & Hopfield, 2003, who showed that simple oscillating network models implemented sensory segmentation). Whatever the case may be, the large increase in fitness in our simulations due to the emergence of oscillations strongly argues in favor of an adaptive function. Moreover, it challenges the idea that oscillations are only a meaningless by-product of cell firing and random idling (cf. Pareti & De Palma, 2004).

A common theme in the oscillation literature is the synchronization between different brain areas (e.g., Seidenbecher et al., 2003; Paré & Collins, 2000; Ritz &

Sejnowski, 1997; Gray et al., 1989). It should be noted that such a synchronization has not been observed in the present simulations. Synchronization requires two, initially independent, generators at different locations that gradually equate their frequency and phase. Here, the oscillations were only generated in one location (the recurrent connections between hidden and context module). Presumably, the oscillation generators evolved before the ability to synchronize them. Therefore, oscillations must have a more basic adaptive function than the one served by synchronization of different oscillations. We think that the functions of oscillations and synchronization are complementary. Once evolution had discovered the fitness value of oscillations, other adaptive functions were “exapted” (Gould, 1991) to the oscillations which were now available at many different locations of the neural network.

We suggest that the main function of neural oscillations is to provide an organism with the ability to switch effectively from one dominant mode of processing and behavior to another. An oscillating network is never completely fixed in one mode and it does not take much change in input to tip the balance over to another mode. Our simulations show that this ability is highly adaptive in realistic environmental conditions when sudden switches are sometimes required. The ability may also solve one of the major problems of competitive networks. Such networks can, through a process of constraint satisfaction, settle in a steady state. For instance, a competitive setup has been successfully applied to the modeling of visual attention (i.e., biased competition; Duncan, 1996; see also Phaf, Van der Heijden, & Hudson, 1990). Competitive networks do not, however, possess a supplementary mechanism to get out of the steady state in which it has settled. In order to start a new process of constraint satisfaction, all the activations in the network need to be reset to zero by the modeler. If the steady state consisted of oscillations, it would, according to our hypothesis, be able to switch to another state in the periodically occurring troughs of low and near-zero activations.

The evolutionary simulations did not yield the working memory capacity we initially aimed for. Recently, however, nonevolutionary simulation studies (Wolters & Raffone, 2008; Raffone & Wolters, 2001) have argued that oscillations are a crucial building block for a working memory capacity. Maintenance in working memory was assumed to result in these models from reverberations due to recurrent connections between prefrontal cortex and inferotemporal cortex. The mechanism that optimized pattern segregation also posed a limit on the number of concurrent reverberations. Future evolutionary simulations could show the emergence of a working memory capacity along the lines of Wolters & Raffone (2008) and Raffone & Wolters (2001), but these simulations would need additional sets of recurrent connections relative to the present simulations, in order to allow for reverberations and synchronization. If a working memory consisting

of oscillating patterns of activation would indeed emerge, this would be an interesting demonstration of an “exaptation” (see Gould, 1991) in evolutionary simulations. The oscillations would then not have arisen as a direct adaptation to create the working memory function, but rather it would have been co-opted for this new function.

In our opinion, evolutionary analyses should focus on neural processing architectures (cf. LeDoux, 1996) responsible for a particular function and not solely on external behavior. Genetically prepared behaviors can be distinguished from learned behaviors by the presence of neural substrates shared by all members of a species. In constructing evolutionary accounts for functions while ignoring the neural substrate, the risk exists of inventing new accounts for learned behaviors that actually lack such an evolutionary basis. They also run the risk of ignoring neural processing architectures that may lead to maladaptive behavior in a specific individual, or in a specific instance, but that still have a net adaptive value. An overgeneralization in the direct pathway may, for instance, lead to snake and spider phobias (Öhman & Mineka, 2001). In an evolutionary cognitive neuroscience view, the phobias require no separate evolutionary account, but are a consequence of the “inertia” of the highly adaptive fear system.

In sum, this study shows that evolutionary simulations may produce new models with unexpected functions. These simulations thus contradict the often heard prejudice that with neural modeling you get what you put into the model. This was possible because the model was actually not build by the modeler but by the simulated evolutionary process. The neural processing architectures that emerged in these simulations are also likely to have evolved in biological neural networks, because the evolutionary algorithm produced them under conditions of which all relevant aspects were implemented as complete as possible. The model that emerges under realistic environmental constraints receives a strong confidence boost, when it is difficult to choose between models on the basis of fit to the empirical data alone. Evolutionary computation, thus, adds automatic model production to the tools of the model builder, and ensures the emergence of the most biologically plausible models.

APPENDIX A

The scent gradient (i.e., the diffusion over distance) obeyed the following equation:

$$\begin{aligned} \delta < \delta_{\text{MAX}}: \quad S &= \frac{1}{1 + \delta} \cdot S_{\text{MAX}} \left(1 - \frac{\delta}{\delta_{\text{MAX}}} \right) \cdot P \\ \delta \geq \delta_{\text{MAX}}: \quad S &= 0 \end{aligned} \quad (1)$$

where:

S = variable scent intensity at the sensor,

S_{MAX} = constant scent intensity at the source (set to 25 at full strength),
 δ = distance between source and sensor (in length units),
 δ_{MAX} = maximal distance at which the source was scented (100 length units),
 P = pattern for the scent type–stimulus combination (predators: 1.0A, 0.5B; plants: 0.5A, 1.0B).

APPENDIX B

To calculate an input-node’s activation, the summed intensity (for all objects within range) of the coupled receptor was squashed between 0 and 1 using the equation:

$$\begin{aligned} \xi \geq 0: \quad \sigma(\xi) &= \frac{\xi}{1 + \xi} \\ \xi < 0: \quad \sigma(\xi) &= 0 \end{aligned} \quad (2)$$

To calculate the activation of all other nodes, the incoming weighted activation was summed:

$$y_{i,t+1} = \sigma \left(\sum_{j=1}^N \omega_{ij} y_{j,t} + \theta_i \right) \quad (3)$$

where:

$y_{i,t}$ = the variable activation of node i at time t ,
 ω_{ij} = the weight on the connection from node j to node i ,
 N = the total number of connections to node i ,
 θ_i = the bias of node i (0.1 in the simulations).

The sigmoid function, used to squash the summation between 0 and 1, was the same as for the input nodes (Equation 2).

APPENDIX C

Offspring was created repeatedly until the population had tripled. The chances of becoming a parent were independent of the number of offspring an agent had previously produced. To create offspring, two agents were selected. The first was the agent with the highest fitness of a random subset (one third of the population size) of agents (i.e., “tournament selection”). The chance of an agent being selected as the second agent was proportional to its fitness and inversely proportional to the Euclidian distance of its genetic weight configuration to the configuration of the first selected agent (i.e., “roulette wheel selection”). Each parent was used to create one offspring. Crossover (i.e., the combination of the weight configurations from both parents) occurred with a probability of .5. The genes between two randomly selected points in the weight array were inherited from the first parent and the rest from the second. After copying and/or combining

the connection weights from the parents, the genetic string was subjected to mutations. The mutation obeyed a Poisson distribution (see Equation 4) with the mutation parameter determining the width of the distribution.

$$M = -0.1 \cdot \log_e \left(\frac{1}{r} - 1 \right) \quad (4)$$

where:

M = the variable mutation size added to a weight,
 r = a random number from a uniform distribution between 0 and 1,

For r is .5, M is zero; if it is smaller, M is negative; and if it is larger, M is positive. The sum of the weight and M was rounded to the range $[-10, 10]$, the weights are set to either -10 or 10 .

After the offspring was created, their fitness was tested. The parents were retested and their results added to their previous scores. After all individuals were measured, their score yielded their chance to remain in the population for another generation cycle. The probability an agent would remain in the population was calculated in three consecutive steps. First, a probability was computed from the fitness relative to the highest and lowest fitness in the population (Equation 5).

$$P_1 = \frac{b \cdot (F_i - F_{\min})}{F_{\max} - F_{\min}} + o \quad (5)$$

where:

P_1 = the probability of survival after the first step,
 F_i = the fitness of individual i ,
 $F_{\min}$ = the lowest fitness of an agent in the generation,
 $F_{\max}$ = the highest fitness of an agent in the generation,
 o = the lower limit of P_1 , that is, a minimal survival chance (0.15),
 b = the range of P_1 values ($1 - o = 0.85$).

In the second step, the probability is adjusted (Equation 6) to keep the population around the initial size.

$$N \geq N_0: \quad P_2 = P_1 \cdot \frac{N_0}{N}$$

$$N < N_0: \quad P_2 = 1 - \left((1 - P_1) \cdot \left(1 - \left(1 - \frac{N}{N_0} \right)^2 \right) \right) \quad (6)$$

P_2 = survival probability after the second step,
 P_1 = survival probability after the first step (Equation 5),
 N = the population size in the last generation,
 N_0 = the initial population size.

In the third step, the probability is multiplied with .99 to ensure that even the fittest individual has some chance of being removed. After removal the cycle restarted with the generation of offspring.

Acknowledgments

We thank Gezinus Wolters, Arman Tajarobi, and the anonymous reviewers for their helpful comments.

Reprint requests should be sent to Bram T. Heerebout, Psychonomics Department, University of Amsterdam, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands, or via e-mail: B.T.Heerebout@uva.nl.

REFERENCES

- Ashby, F. G., Ell, S. W., Valentin, V. V., & Casale, M. B. (2005). FROST: A distributed neurocomputational model of working memory maintenance. *Journal of Cognitive Neuroscience*, 17, 1728–1743.
- Baddeley, A. D. (1986). *Working memory*. Oxford: Oxford University Press.
- Beer, R. D. (1990). Intelligence as adaptive behavior: An experiment in computational neuroethology. *Artificial Intelligence*, 72, 173–215.
- Braitenberg, V. (1984). *Vehicles: Experiments in synthetic psychology*. Cambridge, MA: MIT Press.
- Brody, C. D., & Hopfield, J. J. (2003). Simple networks for spike-timing-based computation, with application to olfactory processing. *Neuron*, 37, 843–852.
- Buzsáki, G., & Draguhn, A. (2004). Neuronal oscillations in cortical networks. *Science*, 304, 1926–1929.
- Cunningham, W. A., Raye, C. L., & Johnson, M. K. (2004). Implicit and explicit evaluation: fMRI correlates of valence, emotional intensity, and control in the processing of attitudes. *Journal of Cognitive Neuroscience*, 16, 1717–1729.
- Darwin, C. (1872/1965). *The expression of the emotions in man and animals*. Chicago: University of Chicago Press.
- Dawson, M. R. W. (2002). From embodied cognitive science to synthetic psychology. In Y. Wang, R. H. Johnston, & M. R. Smith (Eds.), *Proceedings of the First IEEE International Conference on Cognitive Informatics (ICCI'02)* (pp. 13–22). Los Alamitos, CA: IEEE Computer Society.
- den Dulk, P., Heerebout, B. T., & Phaf, R. H. (2003). A computational study into the evolution of dual-route dynamics for affective processing. *Journal of Cognitive Neuroscience*, 15, 194–208.
- Duncan, J. (1996). Cooperating brain systems in selective perception and action. In T. Inui & J. L. McClelland (Eds.), *Attention and performance XVI: Information integration in perception and communication* (pp. 549–578). Cambridge, MA: MIT Press.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14, 179–211.
- Evans, D., & De Back, W. (2002). Synthetic evolutionary psychology. Retrieved 19 May 2005 from www.dylan.org.uk/syntheticEP14.html.
- Francis, A. J. (1973). *Impulse and momentum*. London: Bell.
- Fries, P., Nikolic, D., & Singer, W. (2007). The gamma cycle. *Trends in Neurosciences*, 30, 309–316.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison-Wesley.
- Gould, S. J. (1991). Exaptation: A crucial tool for evolutionary psychology. *Journal of Social Issues*, 47, 43–65.
- Gray, C. M., König, P., Engel, A. K., & Singer, W. (1989). Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338, 334–337.
- Haykin, S. (1999). *Neural networks: A comprehensive foundation*. London: Prentice-Hall International.

- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press.
- Hopfield, J. J., & Tank, W. (1986). Computing with neural circuits: A model. *Science*, 233, 625–633.
- Hutcheon, B., & Yarom, Y. (2000). Resonance, oscillation and the intrinsic frequency preferences of neurons. *Trends in Neurosciences*, 23, 216–222.
- Klausberger, T., & Somogyi, P. (2008). Neuronal diversity and temporal dynamics: The unity of hippocampal circuit operations. *Science*, 312, 53–57.
- LeDoux, J. E. (1986). Sensory systems and emotion: A model of affective processing. *Integrative Psychiatry*, 4, 237–248.
- LeDoux, J. E. (1996). *The emotional brain*. New York: Simon & Schuster.
- Lengyel, M., Huhn, Z., & Érdi, P. (2005). Computational theories on the function of theta oscillations. *Biological Cybernetics*, 92, 393–408.
- Lorenz, K. (1966). Evolution of ritualisation in the biological and cultural spheres. *Philosophical Transactions of the Royal Society of London*, 251, 273–284.
- Michalewicz, Z. (1992). *Genetic algorithms + data structures = evolution programs*. New York: Springer.
- Nilsson, D.-E., & Pelger, S. (1994). A pessimistic estimate of the time required for an eye to evolve. *Proceedings of the Royal Society London, Series B*, 256, 53–58.
- Northcutt, R. G., & Kaas, J. H. (1995). The emergence and evolution of mammalian neocortex. *Trends in Neurosciences*, 18, 373–379.
- Öhman, A., & Mineka, S. (2001). Fears, phobias, and preparedness: Toward an evolved module of fear and fear learning. *Psychological Review*, 108, 483–522.
- Paré, D., & Collins, D. R. (2000). Neuronal correlates of fear in the lateral amygdala: Multiple extracellular recordings in conscious cats. *Journal of Neuroscience*, 20, 2701–2710.
- Pareti, G., & De Palma, A. (2004). Does the brain oscillate? The dispute on neuronal synchronization. *Neurological Sciences*, 25, 41–47.
- Peck, S. L. (2004). Simulation as experiment: A philosophical reassessment for biological modeling. *Trends in Ecology and Evolution*, 19, 530–534.
- Phaf, R. H., Mul, N. M., & Wolters, G. (1994). A connectionist view on dissociations. In C. Umiltà & M. Moscovitch (Eds.), *Attention and performance XV: Conscious and nonconscious information processing* (pp. 725–751). Cambridge, MA: MIT Press.
- Phaf, R. H., Van der Heijden, A. H. C., & Hudson, P. T. W. (1990). SLAM: A connectionist model for attention in visual selection tasks. *Cognitive Psychology*, 22, 273–341.
- Phelps, E. A., & LeDoux, J. E. (2005). Contributions of the amygdala to emotion processing: From animal models to human behavior. *Neuron*, 48, 175–187.
- Raffone, A., & Wolters, G. (2001). A cortical mechanism for binding in visual working memory. *Journal of Cognitive Neuroscience*, 13, 766–785.
- Reyes, A. D. (2003). Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro. *Nature Neuroscience*, 6, 593–599.
- Ritz, R., & Sejnowski, T. J. (1997). Synchronous oscillatory activity in sensory systems; new vistas on mechanisms. *Current Opinion in Neurobiology*, 7, 536–546.
- Rosenblatt, F. (1962). *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*. New York: Spartan.
- Rougeul-Buser, A., & Buser, P. (1997). Rhythms in the alpha band in cats and their behavioural correlates. *International Journal of Psychophysiology*, 26, 191–203.
- Schacter, D. L., Addis, D. R., & Buckner, R. L. (2008). Episodic simulation of future events: Concepts, data, and applications. *Annals of the New York Academy of Sciences*, 1124, 39–60.
- Schaefer, A. T., Angelo, K., Spors, H., & Margrie, T. W. (2006). Neuronal oscillations enhance stimulus discrimination by ensuring action potential precision. *Public Library of Science*, 4, 1010–1024.
- Seidenbecher, T., Laxmi, T. R., Stork, O., & Pape, H.-C. (2003). Amygdalar and hippocampal theta rhythm synchronization during fear memory retrieval. *Science*, 301, 846–850.
- Ulanovsky, N., & Moss, C. F. (2007). Hippocampal cellular and network activity in freely-moving echolocating bats. *Nature Neuroscience*, 10, 224–233.
- Wolters, G., & Raffone, A. (2008). Coherence and recurrency: Maintenance, control and integration in working memory. *Cognitive Processing*, 9, 1–17.